REMARKS

Claims 1-31 remain pending in the application. Reconsideration is respectfully requested in light of the following remarks.

Section 103(a) Rejections:

The Examiner rejected claims 1-5, 7-12, 14 and 17-31 under 35 U.S.C. § 103(a) as being unpatentable over Carre (U.S. Patent 6,282,579) in view of Hamilton et al. (U.S. Patent 5,758,186) (hereinafter "Hamilton"), and claims 6, 13, 15 and 16 as being unpatentable over Carre in view of Hamilton and further in view of Kung et al. (U.S. Patent 6,775,267) (hereinafter "Kung). Applicants respectfully traverse these rejections for at least the reasons presented below.

In regards to claim 1, contrary to the Examiner's contention, **Carre in view of Hamilton fails to teach or suggest a client *generating* a *request* for type information for an attribute or event pertaining to management of one or more managed network objects, wherein each managed network object is a computer programming language object representing one or more devices on a network.** As argued previously, the Examiner erroneously equates any address conversion performed as part of any remote procedure call with specifically *generating a request for type information* for an attribute or event pertaining to management of one or more managed network objects. However, as shown below, merely performing an address conversion does not teach or suggest a *client generating a request* for type information.

Carre pertains to address conversion between CORBA objects and OSI objects (Carre - col. 1, lines 9-19; col. 1, line 59 - col. 2, line 46) and to the transforming of object interfaces column 5, lines 49-59), while Hamilton teaches a system for handling diverse protocols with remote procedure calls (Abstract, col. 1, line 57 – col. 2, line 34). The Examiner cites the Abstract, FIG. 2A, 2B, as well as column 3, line 18 to column 4, line 62 and column 6, lines 10-35 of Carre.

At the Examiner's cited passages, Carre describes how objects interact and communicate using a CORBA infrastructure. In order to permit address interaction between objects that use different addressing modes, Carre's system converts an address type with an address value according to one addressing mode to a corresponding type of another specification language or addressing mode. (Carre, column 1, line 59 - column 2, line 6; column 2, lines 11-14; lines 25-28; and lines 34- 39). Thus, Carre is concerned with *converting address types and object interfaces*, but fails to teach anything regarding *a client generating a request for type information*. Even if combined with Hamilton, Carre's system does not include a client *generating a request for type information*, either at the Examiner's cited passages or elsewhere.

In the Response to Arguments, the Examiner repeats citations from the rejection of claim 1. The Examiner apparently equates any address conversion that may occur during the course of a remote procedure as *a client generating a request* for type information. However, the Examiner's interpretation is simply incorrect. Address conversions that are made during the normal course of other processing do not require *a client generating a request for type information*. Carre in view of Hamilton teaches address conversion, but fails to teach or suggest a client generating a request for type information. Converting address types and object interfaces as part of a protocol conversion, as taught by Carre in view of Hamilton, does not include *a client generating* a *request* for type information.

Carre teaches, even when combined with Hamilton, that client objects request services provided by service objects. Carre further teaches that a client object sends a request message to a service object that contains an operation, a target object, one or more parameters, and, optionally, a request context (Carre, column 3, lines 47-53). **However, Clients in Carre's system do not generate requests for type information**, but instead request a specific service from a particular server object. Carre's system may automatically make any type and address conversions between different object specification languages necessarily to allow the client and server object to communicate.

Thus, in a system resulting from the Examiner's combination of Carre and Hamilton, type and address conversion may take place, **without the agent (client) knowing** that such conversions are performed on the parameters, data, and addresses of various requests between Carre's manager and agent objects.

There is no need in Carre's system for a *client* to *generate a request* for type information for an attribute or event. In fact, the purpose of Carre's teachings appears to be to perform the address type conversion for the client **so that the client does not have to modify it's own interface in order to communicate with an object employing a different addressing mode**. *See, e.g.*, col. 1, line 43 – col. 2, line 6. **Therefore, Carre actually teaches away from *a client generating a request for type information* for an attribute or event.**

Carre in view of Hamilton also fails to disclose *sending the request for type information* generated by a client to an object request broker. The Examiner admits that Carre fails to teach or suggest sending a request for type information to an object request broker and relies on Hamilton, repeating the citations from the rejection of claim 1 (Hamilton, Fig. 1, abstract, column 3, lines 4 – 67 and column 4, lines 14 – 60). However, like Carre, Hamilton does not teach or suggest anything about sending *a request for type information* to an object request broker. Hamilton teaches a system for handling diverse protocols with remote procedure calls. Specifically, Hamilton teaches that method descriptors located within method calls received by a server are specified in a protocol-dependent format. Hamilton also teaches that the method descriptors are compared to other protocol-dependent values and matching values are passed to a protocol-independent processing module that executes the method and returns the reply to the client. See, e.g., Hamilton, column 1, line 57-column 2, line 34).

Applicants have previously argued that data marshalling and method invocations may involve address conversion, but do not involve sending a request (generated by a client) for type information to an object request broker. In the Response to Arguments, the Examiner again refers to Hamilton's subcontract server 58 "performing data

marshalling and other operations of method invocations," without providing any addition response to Applicants' argument. The Examiner specifically refers to "managing operations such as OSI objects on the Agent upon requests from the Manager unit" and cites item M of Fig. 2a from Carre. However, Carre, even if combined with Hamilton, does not describe his manager unit as sending (or receiving) a request (generated by a client) for type information. As noted previously, the Examiner is erroneously equating the automatic type and conversions performed as part of a protocol translation with the specific limitation of a client generating a request for type information.

Furthermore, Carre in view of Hamilton also fails to disclose <u>the client receiving the translated type information</u>. Claim 1 requires, among other things, that the client generate a request for type information for an attribute or event, and that the client receive the translated type information for the attribute or event. Following the Examiner's line of reasoning, agent (A) or manager unit (U) would have to receive the requested type information, translated to IDL. However, the automatic address and/or type conversions performed as part of Carre's protocol translation are performed on communication message sent between Carre's agent and manager units. Carre's automatic conversions are preformed so that an object configured to communicate with one protocol can understand messages from another object configured to communicate using a different protocol. **Any address or other information converted or translated by Carre's system is not returned to the object or unit sending the request.** Instead, the converted data, parameters and address are sent to the receiving object. Any return communication from the receiving object back to the requesting object may also be subject to Carre's protocol translation and therefore Carre's automatic address conversions.

Thus, the converted address and/or other information converted as part of delivering a request from Carre's managing object (M) to Carre's agent object (A) is not then returned to the managing object. Instead, a message sent from M to A in Carre's system would be converted as part of Carre's protocol translation and then sent to A. The converted message is not returned to M. Instead, a separate, return, message from A may

be sent back to M and may also be translated to a different protocol. Thus, Carre's object communication messages, even if combined with Hamilton, cannot be considered request for type information, as recited in claim 1.

Carre in view of Hamilton further fails to disclose a metadata gateway receiving the request for type information from the object request broker. The Examiner admits that Carre fails to teach or suggest a metadata gateway receiving the request for type information from the object request broker and relies on Hamilton. However, Hamilton, even when combined with Carre, fails to mention anything regarding a metadata gateway *receiving a request for type information* from an object request broker. The Examiner, in the response to arguments, again cites Fig. 1, column 3, lines 4 – 67 and column 4, lines 14 – 60 of Hamilton. However, the cited portions do not describe anything regarding a request for type information or about a metadata gateway receiving a request for type information from an object request broker. Instead, the cited passages describe how Hamilton's client stubs pass remote procedure calls to client subcontract processes that, in turn, package the remote procedure calls for transport. On the server side, according to Hamilton, server subcontract processes passes the received package to server stubs that make procedure calls "to execute a specified method of the invoked object." Thus, the cited portions of Hamilton describe a remote procedure call mechanism. Nowhere does Hamilton mention anything regarding a receiving a request for type information from an object request broker. Thus, contrary to the Examiner's assertion, Hamilton combined with Carre does not teach or suggest **a metadata gateway receiving the request for type information from the object request broker.**

Carre in view of Hamilton does not disclose reading the type information from a metadata repository, wherein the type information is stored *in a database format* in the metadata repository. The Examiner cites CMISE/IDL FIG. 3. However, the cited CMISE/IDL is not a metadata repository from which type information is read. In response, the Examiner refers to "using CMISE/IDL fig. 3 as a metadata repository to manage/store the OSI objects translated from type conversion" and also cites column 6, lines 1 – 35 of Carre. Column 6, lines 1 – 35 describe the conversion of ASN types to a

correspondent IDL type. However, Carre teaches that the CMISE/IDL is an "interface unit" that contains the "CMISE object and the services assigned to this object". Carre also teaches that the CMISE object is specified by an IDL interface and acts, and thus appears, like a CORBA object (Carre, column 5, lines 21-39). Thus, the CMISE/IDL cited by the Examiner is a communication interface that allows non-CORBA objects to be interacted with via standard CORBA interfaces. The CMISE/IDL interface unit is clearly not a metadata repository. Moreover, Applicants' claim specifically recites that type information is stored *in a database format* in the metadata repository. Carre's gateway does not store type information in a database format and thus cannot be considered the metadata repository of Applicants' claim.

Carre in view of Hamilton also fails to disclose the client receiving the translated type information for the attribute or event through the object request broker. The Examiner cites column 6, lines 10-35 of Carre. However, this portion of Carre does not describe a client receiving the translated type information for the attribute or event through the object request broker. Instead, as noted above, the cited passage is part of a description of converting objects from ASN to IDL and vice versa as well as caching of converting object instances. As described above, converting CMISE and CORBA objects is not the same as client generating a request to type information, and receiving the translated type information for the attribute or event through an object request broker. The Examiner's cited passage makes no mention of any client receiving translated type information for an attribute or event through an object request broker.

Moreover, clients in Carre's system, even if combined with Hamilton's teachings, do not receive any translated type information through an object request broker. Carre and Hamilton both teach remote procedure call mechanism for executing remote methods over a network. Consequently, clients in Carre's and Hamilton's systems, whether considered singly or in combination, receive the results of the executed method. Clients in Carre and Hamilton do not receive translated type information through an object request broker.

For at least the reasons above, the rejection of claim 1 is not supported by the cited art and removal thereof is respectfully requested. Similar remarks as those above regarding claim 1 also apply to claim 22.

Regarding claim 2, Carre in view of Hamilton does not teach or suggest **translating the type information from the database format to an abstract syntax notation and translating the type information from the abstract syntax notation to the interface definition language**. The Examiner cites column 1, lines 34-55 and column 5, line 60 – column 6, line 21, specifically referring to Carre's teachings regarding using semantic conversions. However, the cited passages, as well as the remainder of Carre, even if combined with Hamilton, only describe converting address values from abstract syntax notation to interface definition language as part of executing a remote procedure call. For example, Carre teaches that an OSI address value is converted to a correspondent IDL type (Carre, column 6, lines 1-2). Please note that Carre teaches that OSI objects are specified in ASN (Carre, column 1, lines 40-42). Thus, the Examiner's cited passages describe converting between ASN and IDL (and the reverse), but fail to mention anything about translating from a database format to an abstract syntax notation.

Additionally, Hamilton, even if combined with Carre, also fails to teach or suggest translating the type information from the database format to an abstract syntax notation and translating the type information from the abstract syntax notation to the interface definition language. Thus, Care and Hamilton, whether considered singly or in combination, do not teach or suggest the limitations of claim 2.

Therefore, the rejection of claim 2 is not supported by the cited art and removal thereof is respectfully requested. Similar remarks apply to claim 23, as well.

Regarding claim 10, **Carre in view of Hamilton fails to teach or suggest a client generating a request to encode type information for an object, attribute, or event pertaining to management of one or more managed network objects, wherein**

each managed network object is a computer programming language object that represents one or more devices on a network. The Examiner cites the Abstract, FIG. 2A and 2B as well as column 3, lines 18-55 and column 5, lines 4-38 of Carre. The Examiner refers to Agent 1 of FIG. 3a, equating Agent 1 to the client of claim 10. However, Carre, even in combination with Hamilton, does not describe, either at the Examiner's cited passages or elsewhere, Agent 1 or any other client, *generating a request to encode type information* for an object, attribute, or event. As noted above regarding the rejection of claim 1, Carre is concerned with providing interface and address type conversions to allow objects specified according to different specification languages, such as ASN and IDL, to communicate and interact with each other by providing interfaces that allow non-CORBA objects to appear as CORBA objects to the outside world. Thus, Carre is concerned with automatic conversions as part of protocol translation during communication between two objects. Hamilton is also concerned with handling multiple protocols when making remote procedure calls.

Clients in Carre's system **do not generate requests to encode type information** for objects, attributes or events. Instead, client objects in Carre's system invoke services provided by server objects, via a standard CORBA interface (Carre, column 1, lines 9-19; column 1, line 59-column 2, line 46; column 5, lines 49-59). Carre's address conversion has nothing to with a client *generating a request to encode type information for an object, attribute or event*. Instead, Carre's address type conversion is performed as part of communicating with CMISE object that appear as CORBA objects. Nowhere does Carre mention a client *generating a request to* encode type information. Instead, Carre's interface unit converts object instances and object instance values into an IDL type to allow communication between CORBA objects and CMISE objects. Nowhere does Carre, even if combined with Hamilton, mention a client generating a request to encode type information for an object attribute or event.

Additionally, **Carre in view of Hamilton does not teach or suggest sending the request to encode type information to an object request broker.** The Examiner cites ORB and CMISE Gateway of FIG. 3a. However, Carre, even if combined with

Hamilton, does not describe the ORB of FIG. 3a as receiving a request to encode type information from a client. Instead, Carre teaches object request brokers provide an infrastructure that enables objects to communicate in a distributed environment such that "it makes no difference" to the requesting objects in which computer system or in which form the target object is implemented (Carre, column 3, lines 56-63). Carre also teaches that a requesting object sends a request message to an object request broker and that the object request broker routes the request message to the target object (Carre, column 3, line 64-column 4, line 6). Thus, Carre teaches that object request brokers route request messages from a request object to the target object. Carre in view of Hamilton does not mention anything about object request brokers receiving requests to encode type information from a client.

Carre in view of Hamilton also fails to teach or suggest a metadata gateway receiving the request to encode type information from the object request broker. The Examiner admits that Carre fails to teach or suggest a metadata gateway receiving the request to encode type information from the object request broker and relies on Hamilton, citing Fig. 1, column 3, lines 4 – 67 and column 4, lines 14 – 60. As noted above, the cited passages describe how Hamilton's client stubs pass remote procedure calls to client subcontract processes that, in turn, package the remote procedure calls for transport. On the server side, according to Hamilton, server subcontract processes passes the received package to server stubs that make procedure calls "to execute a specified method of the invoked object." Thus, the cited portions of Hamilton describe a remote procedure call mechanism.

Nowhere does Hamilton, even in view of Carre, teach or suggest a metadata gateway receiving the request to encode type information from the object request broker. In fact, **nowhere does Hamilton mention anything about any sort of request to encode type information at all**. Hamilton's remote procedure call mechanism handles multiple, diverse, protocols, but does not, even when combined with Carre's remote procedure call mechanism, pertain to a metadata gateway receiving a request to encode type information from an object request broker.

Carre in view of Hamilton further fails to teach or suggest storing the type information in a metadata repository, where the type information is stored *in a database format* in the metadata repository. The Examiner does not cite any portion of Carre or Hamilton regarding this limitation of claim 10. In fact, the Examiner does not mention this limitation in the rejection of claim 10 at all.

The Examiner has previously cited CMISE/IDL of FIG. 3 and column 6, lines 10-35, specifically referring to Carre's conversion of address types from OSI types. However, these portions of Carre refer to the caching of structures, such as object instance values and object reference pairs for addresses already in an entity. Caching of object values and references is not the same as storing type information in a database format in a metadata repository. Additionally, as noted above regarding the rejection of claim 1, the CMISE/IDL interface unit of Carre is clearly a communication interface that allows non-CORBA objects to be interacted with via standard CORBA interfaces (Carre, column 5, lines 21-39). The CMISE/IDL interface unit is clearly not a metadata repository. Moreover, Carre does not describe or even mention anything regarding storing type information in the CMISE/IDL interface unit. Similarly, Hamilton also fails to teach or suggest storing type information in a metadata repository. Thus, Carre in view of Hamilton fails to teach or suggest storing the type information in a metadata repository.

As discussed above, the rejection of claim 10 is not supported by the cited art and removal thereof is respectfully requested. Similar remarks as those above regarding claim 10 apply to claim 27 as well.

Regarding claim 11, **Carre in view of Hamilton fails to disclose translating the type information from the interface definition language to an abstract syntax notation and translating the type information from the abstract syntax notation to the database format**. The Examiner cites column 1, lines 34-55 and column 5, line 60 to column 6, line 21, of Carre, specifically referring to Carre's use of semantic

conversions. However, the cited passages, as well as the remainder of Carre, even if combined with Hamilton's teachings, only describe converting address values from abstract syntax notation to interface definition language and from interface definition language to abstract syntax notation. For example, Carre teaches that an OSI address value is converted to a correspondent IDL type (Carre, column 6, lines 1-2). Please note that Carre teaches that OSI objects are specified in ASN (Carre, column 1, lines 40-42). Thus, the Examiner's cited passages describe converting between ASN and IDL (and the reverse), but fail to mention anything about translating from an abstract syntax notation to a database format.

For at least the reasons above, the rejection of claim 11 is not supported by the cited art and removal thereof is respectfully requested. Similar remarks also apply to claim 28.

Regarding claim 14, Carre in view of Hamilton fails to teach or suggest a metadata repository comprising metadata concerning object classes for a plurality of managed objects, wherein the metadata comprises information expressed in a database format, and wherein the managed objects are computer programming language objects corresponding to managed devices on a network. The Examiner cites CMISE/IDL of FIG. 3a as well as the abstract, FIGs 2A, 3A, column 3, lines 18-55 and column 5, lines 4-38 of Carre. However, as noted above, regarding the rejections of claim 1 and 10, the CMISE/IDL interface unit cited by the Examiner is clearly a communication interface that allows non-CORBA objects to be interacted with via standard CORBA interfaces (Carre, column 5, lines 21-39). **The CMISE/IDL interface unit is clearly not a metadata repository**. Moreover, Carre, even if combined with Hamilton, does not describe or even mention anything regarding storing metadata concerning object classes in the CMISE/IDL interface unit. Nowhere does Carre, even when combined with Hamilton, describe anything about storing metadata concerning object classes in a metadata repository.

Carre in view of Hamilton also fails to teach or suggest a metadata gateway coupled to the metadata repository and to an object request broker, where the metadata gateway is operable to send and receive metadata from the database, where the metadata gateway provides translation of the metadata to and from the database format and an interface definition language. The Examiner admits that Carre fails to teach this limitation of claim 14 and relies on Hamilton, again citing Fig. 1, column 3, lines 4 – 67 and column 4, lines 14 – 60. As noted above, the cited passages describe how Hamilton's client stubs pass remote procedure calls to client subcontract processes that, in turn, package the remote procedure calls for transport. On the server side, according to Hamilton, server subcontract processes passes the received package to server stubs that make procedure calls "to execute a specified method of the invoked object." Thus, the cited portions of Hamilton describe a remote procedure call mechanism.

However, Hamilton's remote procedure call mechanism, even if combined with Carre's teachings, does not involve, nor pertain to, a metadata gateway that provides translation of metadata to and from a database format and an interface definition language. Instead, Hamilton teaches that protocol-dependent values that match a method descriptor are located in a database and passed to a server stub that executes the method indicated by the method descriptor (Hamilton column 1, line 56 – column 2, line 34; FIG. 3; column 5, lines 20-39; and column 6, lines 23-45). Please note that Hamilton does not teach that any translation of metadata to and from a database format and an interface definition language. Instead, Hamilton, even when combined with Carre, teaches that values (e.g., protocol-dependent values) are located in a database and passed without translation to server stub processes for use when executing an indicated method. Nothing about Hamilton's system involves a metadata gateway providing translation of metadata to and from a database format and an interface definition language.

Thus, the rejection of claim 14 is not supported by the cited art and removal thereof is respectfully requested.

Regarding claim 17, **Carre in view of Hamilton fails to disclose wherein the metadata gateway includes a library of data types expressed in an abstract syntax notation, a plurality of object types, wherein each object type includes one or more of the data types from the library of data types.** The Examiner cites, FIG. 2A, 3A, column 4, lines 7-62 and column 5, line 39 to column 6, line 35 of Carre. However, the Examiner has failed to cite any portion of Carre (or Hamilton) that describes a library of data types *expressed in an abstract syntax notation*. Additionally, Carre's CMISE Gateway, which the Examiner equates to the metadata gateway of Applicants' claims, does not include a library of data types. Carre makes no mention of his CMISE Gateway including a library of data types expressed in an abstract syntax notation. Instead, Carre describes his CMISE Gateway as providing address type conversion between two different object interface specifications, such as between IDL and C++. Merely providing address type conversion does not imply including a library of data types and a plurality of object type, each including one or more of the data types from the library. The Examiner's interpretation of Carre's CMISE Gateway is incorrect.

Additionally, Carre in view of Hamilton also fails to teach or suggest wherein the metadata gateway includes an interface to the plurality of object types, wherein the interface is operable to provide one or more clients with access to the metadata as expressed in the interface definition language. The Examiner's cited passages of Carre do not describe the CMISE Gateway, which the Examiner equates to the metadata gateway of Applicants' claims, as including an interface to a plurality of object types, where the interface provides clients with access to the metadata. Instead, the Examiner's cited passages teach that Carre's CMISE Gateway translates objects and object address types from ASN to IDL and from IDL to ASN. Nowhere does Carre describe that the CMISE Gateway includes an interface providing clients access to metadata.

Hamilton also fails to describe or teach anything that overcomes Carre's failure to teach the subject matter on which the Examiner relies. Thus, Carre and Hamilton, whether considered singly or in combination, do not teach or suggest a metadata gateway that includes a library of data types expressed in an abstract syntax notation, a plurality of

object types, wherein each object type includes one or more of the data types from the library of data types. Carre and Hamilton, whether considered singly or in combination, also fail to teach or suggest that the metadata gateway includes an interface to the plurality of object types, wherein the interface is operable to provide one or more clients with access to the metadata as expressed in the interface definition language.

Thus, for at least the reasons above, the rejection of claim 17 is not supported by the cited art and removal thereof is respectfully requested.

Applicant also asserts that numerous ones of the dependent claims recite further distinctions over the cited art. However, since the rejection has been shown to be unsupported for the independent claims, a further discussion of the dependent claims is not necessary at this time.

CONCLUSION

Applicants respectfully submit that the application is in condition for allowance, and prompt notice to that effect is respectfully requested.

If any fees are due, the Commissioner is authorized to charge said fees to Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C. Deposit Account No. 501505/5181-46200/RCK.

Respectfully submitted,

/Robert C. Kowert/
Robert C. Kowert, Reg. #39,255
Attorney for Applicants

Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C.
P.O. Box 398
Austin, TX 78767-0398
Phone: (512) 853-8850

Date: August 6, 2007